



# STEMsalabim: A high-performance computing cluster friendly code for scanning transmission electron microscopy image simulations of thin specimens



Jan Oliver Oelerich\*, Lennart Duschek, Jürgen Belz, Andreas Beyer, Sergei D. Baranovskii, Kerstin Volz

Faculty of Physics and Material Sciences Center, Philipps-Universität Marburg, Germany

## ARTICLE INFO

### Article history:

Received 2 November 2016

Revised 17 January 2017

Accepted 5 March 2017

Available online 10 March 2017

### MSC:

82-04

78A15

### Keywords:

Multislice simulations

Electron scattering factors

MPI

Phonons

## ABSTRACT

We present a new multislice code for the computer simulation of scanning transmission electron microscope (STEM) images based on the frozen lattice approximation. Unlike existing software packages, the code is optimized to perform well on highly parallelized computing clusters, combining distributed and shared memory architectures. This enables efficient calculation of large lateral scanning areas of the specimen within the frozen lattice approximation and fine-grained sweeps of parameter space.

© 2017 Elsevier B.V. All rights reserved.

## 1. Introduction

Since the early 1980s scanning transmission electron microscopy (STEM) has evolved into an indispensable tool for the structural analysis of thin crystalline specimen [1]. Using high-angle annular dark field (HAADF) detectors, chemically sensitive scattering of the focused electron probe can reveal valuable information about the sample, such as chemical composition, strain fields, thicknesses, etc [2–5]. In order to extract quantitative information from STEM images, however, often it is still necessary to compare measured data to computer simulations of the interaction between incident electron beam and the specimen.

Attempts to simulate such electron-specimen interaction numerically go back to the mid-1970s, when Goodman and Moodie [6] first implemented the multislice algorithm proposed earlier by Cowley and Moodie [7]. Since then, the multislice approach evolved into the most practical approach for the simulation of transmission electron microscopy (TEM) and STEM images and is widely applied today for the analysis of experimental (S)TEM im-

ages [2]. A brief description of the multislice algorithm is given in Section 2.

Although the basic algorithm of the multislice method is relatively simple and can be carried out very efficiently using highly optimized discrete fourier transform (DFT) packages [1], a typical STEM image simulation is still computationally demanding: For each lateral scan position of the focused electron probe, the full interaction between the incident electron wave and the specimen has to be simulated. For instance, modeling a STEM image with only 128 pixels  $\times$  128 pixels scan points requires more than 16k multislice simulations. This number grows significantly when taking into account thermal diffuse scattering (TDS) by applying the frozen lattice approximation [8] (see Section 2), requiring all simulations to be repeated a number of times with thermally displaced atomic positions. In order to further account for chromatic aberrations one needs to average over multiple probe defoci [9–12], each of which requires another repetition of the complete set of simulations. For typical values of 10 frozen lattice configurations (the necessary number for convergence depends on sample thickness) and a defocus series consisting of 7 defocus values, the total number of multislice simulations required for the 128 pixels  $\times$  128 pixels STEM image easily exceeds one million. Needless to say, systematic parameter variations or simulations of larger supercell sizes further drastically increase the computational demand.

\* Corresponding author.

E-mail address: [jan.oliver.oelerich@physik.uni-marburg.de](mailto:jan.oliver.oelerich@physik.uni-marburg.de) (J.O. Oelerich).

Although there are many existing implementations of the multislice algorithm for the simulation of STEM images [1,13–23], almost all of them are designed for desktop computers with a single or very few processors or don't comply with common parallelization standards and are therefore not suited for realistic large-scale simulations, parameter sweeps, or iterative optimization. Particularly surprising is the lack of a freely available code for massive parallelization on high-performance computing (HPC) clusters, even though the many necessary multislice simulations for a single STEM image are independent from each other and therefore perfectly suited for efficient parallelization.

In this article we present *STEMsalabim*, a new implementation of the multislice algorithm that is built specifically for concurrent calculations on many computing nodes. The package uses a combination of the message passing interface (MPI) and (POSIX) threads for efficient parallelization while maintaining a small memory footprint, a feature that is not present in existing software packages but crucial for the simulation of large images. *STEMsalabim* supports TDS via the frozen lattice approximation [8], is able to simulate chromatic aberrations by means of defocus series [9–12], and implements lens aberrations of focused electron beams in STEM up to fifth order [24].

In Section 2 we summarize the multislice frozen lattice algorithm as implemented in *STEMsalabim* and give implementation details. Simulation results and comparison with experimental images are shown in Section 3, benchmarks and parallel scaling behaviour of the package are summarized in Section 4. Conclusions and acknowledgements are given in Section 5 and acknowledgement.

## 2. Algorithm and implementation

*STEMsalabim* implements the standard multislice algorithm as explained in great detail in the book of Kirkland [1]. In this approximation, the supercell is divided into slices of some thickness  $dz$  perpendicular to the direction of the electron beam  $z$ . Passing through a slice, the electronic wave function is first modified by interaction with the 2D projected potential of the slice's atoms, and is then propagated by a Fresnel propagator through empty space of thickness  $dz$ . This process is repeated for all slices in the supercell. Afterwards, the wave intensity is integrated and collected over some momentum range to simulate an HAADF ring detector. The multislice algorithm is repeated for each scan position of the incident focused electron probe, for varying atomic positions in order to resemble TDS, and for different defocus values to account for chromatic aberrations.

### 2.1. Multislice method

Let us briefly repeat the most important equations required for the multislice method as described by Kirkland [1].

The supercell is split into  $N$  consecutive slices of thickness  $dz$  perpendicular to the beam direction  $z$ . The projected potential  $v_s(x, y)$  for each slice  $s$  is evaluated as the cumulative atomic potential of all atoms  $i$  in the slice:

$$v_s(x, y) = \sum_i \int V_i(x - x_i, y - y_i, z) dz. \quad (1)$$

The atomic potentials  $V_i(|\vec{r}|)$  for the elements  $i$  are calculated from the atomic scattering amplitudes  $f_i(\vec{k})$  by employing the first Born approximation

$$V_i(|\vec{r}|) \propto \mathcal{F}^{-1}(f_i(\vec{k})), \quad (2)$$

where  $\mathcal{F}^{-1}$  is the inverse Fourier transform. The scattering amplitudes  $f_i(\vec{k})$  can be parametrized following Ref. [1] using tabulated

values for the element specific parameters  $a_{i,j}$ ,  $b_{i,j}$ ,  $c_{i,j}$ , and  $d_{i,j}$ :

$$f_i(\vec{k}) = \sum_{j=1}^3 \frac{a_{i,j}}{|k|^2 + b_{i,j}} + \sum_{j=1}^3 c_{i,j} \exp(-d_{i,j}|k|^2). \quad (3)$$

Note, that finding correct parametrizations of electron and X-ray scattering factors for a wide angle range is a field of research itself. With *STEMsalabim*, we stick to the parameters tabulated in Ref. [1], which are mostly based on the parametrizations published by Doyle and Turner [25] and Doyle and Cowley [26] but with a different basis (Lorentzian and Gaussian as opposed to the Gaussian-only basis of Doyle and Turner [25]). We are aware of more sophisticated ways to parametrize the scattering factors [27] and are considering them for future versions of the package. However, the one of Kirkland [1] works quite well and yields the same results as the *STEMsim* software [13].

From the slice potential given in Eq. (1), a weak phase object can be calculated for each of the  $N$  slices as

$$t_s(x, y) = \exp[i\sigma v_s(x, y)], \quad (4)$$

with the interaction parameter

$$\sigma = \frac{2\pi\gamma m_0 e \lambda}{h^2}. \quad (5)$$

Here,  $m_0$  is the rest mass and  $e$  the charge of an electron,  $\gamma$  is the Lorentz factor, and  $h$  is Planck's constant.  $t_s(x, y)$  is often referred to as the slice's *transmission function*. After the electron wave is modified by the transmission function, it is propagated to the next slice by applying a Fresnel propagator

$$p(k_x, k_y, dz) = \exp[-i\pi |k|^2 \lambda dz] \quad (6)$$

given in frequency space.

With Eqs. (4), (6), the total interaction between an incident wave function  $\Psi_s(x, y)$  and slice  $s$  is given by

$$\Psi_{s+1}(x, y) = \mathcal{F}^{-1}\{p(k_x, k_y, dz)\mathcal{F}[t_s(x, y)\Psi_s(x, y)]\}, \quad (7)$$

which is iteratively applied for all slices of the supercell until the wave hits the detector (i.e., until the last layer of the supercell was passed).

The incident probe wave function  $\Psi_0(x, y)$  for a focused electron beam resting at the scan position  $(x_p, y_p)$  is given by

$$\Psi_0(x, y, x_p, y_p) = B\mathcal{F}^{-1}(A(k_x, k_y) \exp[-i\chi(k_x, k_y) + 2\pi i(k_x x_p + k_y y_p)]), \quad (8)$$

with a normalization constant  $B$ , the aperture function  $A(k_x, k_y)$ , and the aberration phase error  $\chi(k_x, k_y)$ . The aperture function  $A(k_x, k_y)$  is unity for all  $\lambda|k| < \alpha_{\max}$  and zero outside, with  $\alpha_{\max}$  being the maximal angle of the objective aperture. The aberration phase error  $\chi(\vec{k})$  is given by

$$\chi(k_x, k_y) = \frac{\pi\alpha^2}{\lambda} \left( -\Delta f + C_a \cos(2\phi - 2\phi_a) + \frac{1}{2}C_s\alpha^2 + \frac{1}{3}C_5\alpha^4 \right). \quad (9)$$

In Eq. (9),  $\alpha = \lambda|k|$  is the polar angle,  $\phi$  the azimuthal angle, and  $\delta f$ ,  $C_a$ ,  $\phi_a$ ,  $C_s$ ,  $C_5$  are the aberration coefficients defocus, two-fold astigmatism, two-fold astigmatism azimuthal angle, and spherical aberrations in third- and fifth-order, respectively.

Note, that both the transmission functions  $t_s(x, y)$  as well as the atomic potentials  $v_s(x, y)$  should be symmetrically bandwidth limited to  $\min(k_{x, \max}, k_{y, \max})$  in order to ensure cylindrical symmetry inherent to electron microscopes [1].

### 2.2. Algorithm

The multislice method as outlined above yields the probe wave function after interaction with the supercell for a single lateral

probe position  $(x_p, y_p)$  and a single configuration of atomic coordinates  $\vec{r}_i$ . In order to simulate a full STEM image, this simulation needs to be repeated for the probe position at all scan points, for varying atomic coordinates in order to account for TDS in the frozen lattice approximation, and for different defoci  $\Delta f$  of the probe (see Eq. (9)) to include chromatic aberrations. The full algorithm is then as follows:

1. Displace atomic coordinates randomly following a Gaussian probability distribution (Einstein solid) to include TDS.
2. Divide the sample into  $N$  slices of thickness  $dz$  and calculate transmission functions  $t_s(x, y)$  using Eqs. (2) and (4).
3. Generate the probe wave function  $\Psi_0(x, y, x_p, y_p)$  with Eq. (8).
4. For each slice  $s$ , calculate  $\Psi_{s+1} \leftarrow \Psi_s$  by applying Eq. (7).
5. Integrate  $|\Psi|$  for the appropriate angle range to simulate the HAADF detector.
6. Repeat steps 3–5 for each scan point  $(x_p, y_p)$ .
7. Repeat steps 1–6 for each frozen lattice configuration.
8. Repeat steps 1–7 for the each defocus  $\Delta f$  of the defocus series to model chromatic aberrations.
9. Appropriately average over frozen lattice configurations and defoci.

### 2.3. Implementation details

As shown in the introduction, a realistic STEM image simulation easily requires millions of single multislice simulations. Luckily, by parallelization and by reusing precalculated objects, one can reduce the computational demand significantly. In this section, we discuss some implementation details regarding MPI and threaded parallelization and a few caching mechanisms that are implemented in STEMsalabim.

The principle interaction between the supercell and the wave function  $\Psi(x, y)$  is via each slice's transmission function  $t_s(x, y)$  as described in Eq. (7), which, in turn, is built from the involved elements' atomic potentials  $V_i(|\vec{r}|)$  via Eq. (1). While the atomic coordinates change between frozen lattice configurations and between slices, the atomic potentials need to be calculated only once for the distances  $|\vec{r}| < r_{\text{cut}}$ . Therefore, it is good practice to precalculate and cache  $V_i(|\vec{r}|)$  for all elements involved in the simulation. The same holds true for the Fresnel propagator from Eq. (6), as it is constant as long as the slice thickness  $dz$  is kept unchanged.

The transmission function  $t_s(x, y)$  for each slice  $s$  can be cached and reused within the same frozen lattice configuration for all STEM probe positions  $(x_p, y_p)$ , since the slice potentials depend solely on the atomic coordinates. This requires the loop over probe positions to be carried out within the loop over frozen lattice configurations, as indicated in the algorithm in Section 2.2. Note, that caching the complex valued discretized transmission functions for all slices requires significant amounts of memory when calculating large lateral scan areas and/or thick supercells.

In principle, the transmission functions could be reused also for different defocus values  $\Delta f$  when simulating a defocus series, as these only influence the probe wave function  $\Psi_0$  given by Eq. (8). However, it proved useful to generate a new set of configurations for each defocus value for convergence of the frozen lattice TDS method. (We consider a simulation as converged when the intensity differences between identical atomic columns become sufficiently small.) Generating new coordinates for each defocus allows significant reduction of required frozen lattice configuration to average over.

As mentioned above, most of the multislice iterations required for a full STEM image simulation are independent from each other and can therefore be parallelized very efficiently. In STEMsalabim, we opted for a hybrid parallelization scheme using MPI for com-

munication between machines and a shared-memory threading architecture for parallelization within one machine. This scheme has some important benefits over pure MPI or threaded parallelization and is often applied for memory critical software [28]. First of all, message passing (MPI) allows us to parallelize across computers, which is crucial for running the program on highly concurrent HPC clusters. Using threads instead of MPI processes for concurrency within one machine greatly reduces the required memory, as the same set of transmission functions  $t_s(x, y)$  can be used via shared memory for parallel multislice simulations. Moreover, the transmission functions for all slices can be calculated in parallel without the need of communicating large chunks of data between MPI processes, which is beneficial for performance. Second of all, reducing the number of involved MPI processes also reduces network activity when reporting simulation results to the MPI master process. Instead, result packages are gathered on each machine independently and are then passed to the master process as a single chunk of data for post-processing and input/output (I/O).

The basic steps of the parallel STEM image simulation as carried out by STEMsalabim, complementary to the algorithm Section 2.2, are as follows:

1. The master MPI process reads in atomic coordinates, randomly generates frozen lattice configurations and communicates the coordinates to the other MPI processes.
2. Each worker MPI process slices the specimen and calculates and caches atomic potentials  $V_i(x, y)$ , transmission functions  $t_s(x, y)$ , and the Fresnel propagator  $p(k_x, k_y, dz)$ .
3. Each worker MPI process queries the master for a work package of some probe positions  $(x_p, y_p)$ , carries out the multislice calculations concurrently by threaded parallelization, and sends the packaged results back to the MPI master. This is repeated until all probe scan points  $(x_p, y_p)$  are processed.
4. Steps 1 - 3 are repeated for all frozen lattice configurations and defocus values  $\Delta f$ .

All processing of results, management of the queue of multislice simulations, and all I/O is carried out by the main thread of the master MPI process to prevent race conditions. The precision of all calculations can be chosen as single or double, depending on the desired level of accuracy.

Since the multislice method essentially consists only of a series of products and convolutions of discretized complex-valued wave functions, the bottle neck of the simulation are the forward and inverse fourier transforms. Fortunately, highly optimized implementations of the fast fourier transform (FFT) exist, that are the working horses of the STEM image simulations. The FFTs in STEMsalabim are carried out by the widely-applied FFTW library. While FFTW is written to run on conventional central processing units (CPUs), it would be easy to replace it with FFT codes that are capable of running on graphics processing units (GPUs), which would result in additional significant performance gain [14].

The input for a simulation with STEMsalabim consists of two files: One file describes the cell dimensions and contains (equilibrium) positions, elements and mean square displacements of all atoms present in the supercell. The second file contains all necessary configuration values for the simulation, e.g., characteristics of the microscope and detector, output file name, etc. Some configuration values can be overridden via command line parameters upon execution of the program.

Simulation results are stored as binary data using NetCDF, which is based on the HDF5 library. This has several advantages over the usual approach of storing data in structured ASCII files. First, the data volume is much smaller due to binary storage and is further compressed by HDF5. This is particularly useful for STEM image simulations where the amount of data produced in simulations can be large. Second, all results, simulation parameters, and

runtime information can be stored in a single result file, which is a much more portable approach than distributed ASCII files. Third, NetCDF/HDF5 bindings exist for all popular programming languages (MATLAB, for example, ships functions for reading binary HDF5 data) and data can be read selectively from the file, avoiding large memory requirements when processing big output files.

Other libraries used by STEMsalabim include libConfig for reading in simulation parameters, various of the Boost C++ libraries for utility functions, and CMake for the build process.

STEMsalabim is written in modern C++ using features of the latest language standards C++11 and C++14. The target platforms of the package are Unix based systems, however, it should work just as well on Microsoft Windows operating systems. The package can be used with both OpenMPI and Intel MPI implementations. Compilation was successfully tested with the GNU Compiler Collection (GCC), LLVM clang, and the Intel Compiler Collection (ICC).

### 3. Results and comparisons

Correctness of images simulated with our package was checked and confirmed by comparison to the well-established STEMsim code [13]. In what follows, we showcase a few exemplary STEM image simulations carried out with STEMsalabim and compare some of the results to experimentally obtained images. The chosen examples show the necessity of TDS, chromatic aberrations, and large lateral supercell sizes in order to reach sufficient accuracy in the simulated images. Runtimes and performance of the simulations are discussed in Section 4.

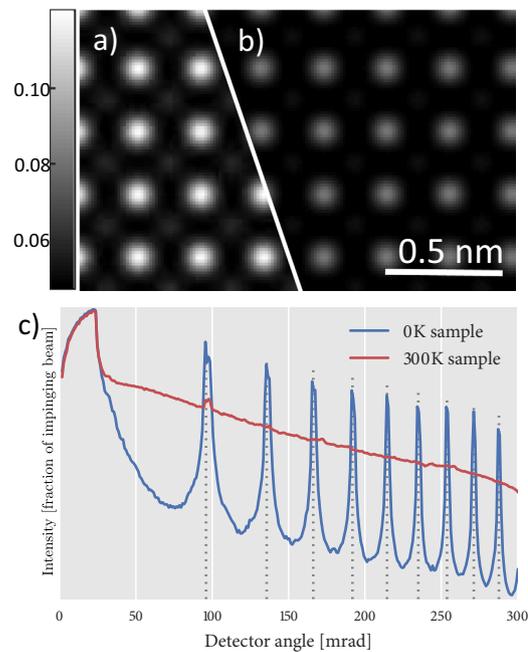
All shown simulations were simulated with an acceleration voltage of 200 kV, a numerical aperture of 24 mrad, and spherical aberration coefficients  $C_s = 2 \mu\text{m}$  and  $C_5 = 5 \text{ mm}$ . These parameters are chosen to match our microscope (JEOL JEM 2200). As astigmatism is corrected, it was disabled in the simulations. The detector angle range used to generate the micrographs was in all cases 74 mrad to 174 mrad.

#### 3.1. GaP at 0K and room temperature

Let us start with a simple image simulation of pure gallium phosphide (GaP), at  $T = 0\text{K}$  (i.e., all atoms rest at their equilibrium lattice positions) and at room temperature ( $T = 300\text{K}$ , by including TDS via the frozen lattice approximation). By comparison of the two simulations we show the significant effects that TDS have on STEM images.

The simulated STEM images for the integrated detector angle range 72 mrad to 172 mrad are shown in Fig. 1(a) and (b), their dependences on detector angle  $\alpha = |\vec{k}|\lambda$  is plotted in (c), along with the theoretical positions of the nine first higher order Laue zone (HOLZ) peaks. For both temperatures, a series of 7 defocus values distributed around  $\Delta f = -2 \text{ nm}$  was calculated and averaged, similar to Refs. [12,29]. The simulated supercell had a lateral size of  $9 \times 9$  unit cells, of which the center square of  $3 \times 3$  unit cells was chosen as scanning area, and a thickness of about 50 nm. For the 300K simulation, TDS was taken into account by calculating and averaging 15 configurations of atomic positions, according to the Einstein approximation.

It is clearly visible that including TDS smears out the atomic columns and reduces collected intensity. The homogeneous brightness and shape of the atomic columns indicates sufficiently large number of frozen lattice configurations for TDS. The intensity dependence on detector angle  $\alpha$  in the diffraction plane of the images, shown in Fig. 1(c), exhibits a clear signature of the HOLZ peaks at their expected positions for the 0K sample. The first order Laue zone (FOLZ) is still visible as a small hump in the curve of



**Fig. 1.** Simulated gallium phosphide (GaP) STEM images at zero (a) and room temperature (b), beam direction [001]. The simulated scanning area was  $3 \times 3$  unit cells, part of which is cropped in the upper figures. The dependences of the images on the detector angle ( $\alpha = |\vec{k}|\lambda$ ) are shown in (c), with the expected positions of the higher order Laue zone (HOLZ) peaks up to 9th order marked by dotted vertical lines. Intensity values are displayed as fraction of impinging beam.

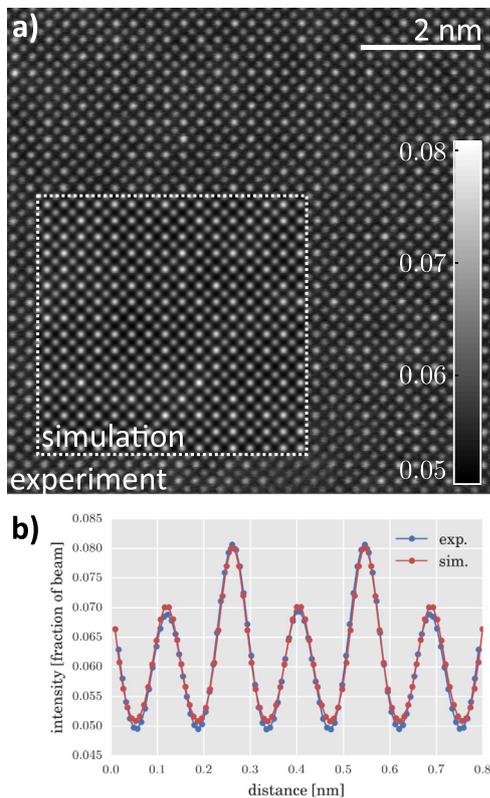
the 300K sample. While the intensity of the former is almost exclusively collected at the HOLZ reflections, the latter sample shows a smooth, decaying angular dependence over the whole detector angle range.

Note, that the used number of 15 frozen lattice configurations per each of the 7 defoci results in 105 independent vibration configurations of the atomic positions. The homogeneity of the shape and intensity of the atomic columns indicates that generating new frozen lattice configurations for each defocus reduces the number of necessary configurations for a single defocus, as 15 configurations are usually not enough for convergence.

#### 3.2. GaAsBi simulation in comparison to experiment

In Fig. 2, we show a STEM image of an approximately 32 nm thick gallium arsenide bismide (GaAsBi) sample with a bismuth (Bi) concentration of 4.8%, taken in a JEOL JEM 2200 TEM operating at 200 kV. A simulated GaAsBi image with equal sample and microscope parameters is shown in an overlay. The atomic positions of the supercell were relaxed prior to the simulation by valence force field (VFF) methods [30]. A series of 7 defoci with 10 frozen lattice configurations each was calculated to obtain the simulation results.

The excellent agreement between the images, both qualitatively and regarding background, group III, and group V intensities, is only possible by including chromatic aberrations. The group V columns with finite Bi content are nicely visible as bright spots in both results, with correct relative intensities to columns without Bi. Both the experimental and the simulated image are shown with the same colorbar, with intensities given in fractions of the impinging electron beam. The main differences left between experimental and simulated image are fly-back errors of the scanning microscope, that manifest as horizontal distortions in the experimental STEM image.



**Fig. 2.** (a) Experimentally obtained STEM image of an approximately 32 nm thin gallium arsenide (GaAs) sample with 4.8% bismuth (Bi). A corresponding sample simulated with STEMsalabim is shown as an overlay framed by a dotted, white line. Intensity values are displayed as fraction of impinging beam. In (b), diagonal line scans over both the simulated and experimentally obtained image are shown. The group III and group V peaks are clearly visible and exhibit quantitative agreement.



**Fig. 3.** STEM image simulation of a GaP/gallium arsenide (GaAs) heterojunction. The supercell dimensions were  $55.88 \text{ nm} \times 1.37 \text{ nm} \times 24.68 \text{ nm}$  in  $x$ ,  $y$ , and beam direction, respectively. Scanning in  $y$  direction was confined to only 60% of the sample size to rule out boundary effects. The simulated image is continued periodically to five times its original size in  $y$  direction. The original supercell is framed by a red box. Below the image simulation, the intensity of the group III columns across the interface is plotted. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

### 3.3. Large simulation of GaP/GaAs heterojunction

A third example is displayed in Fig. 3, a simulation of a heterojunction between GaP and gallium arsenide (GaAs). The supercell morphology was relaxed prior to the simulation by means of finite element (FE) simulation in order to include lattice mismatch effects between the two materials. STEM image simulations allow for tracing the group III and V column intensity across the junction and thereby relate intensity to interface relaxation. In Fig. 3, the intensity on group III columns across the junction is plotted below the simulated STEM image.

It is clear from the plot, that a reduction of the group III intensity in the GaP bulk of up to 25% is induced by the lattice mismatch. The intensity recovers to its GaP bulk value more than 15 nm away from the junction, clearly showing the long-range nature of such effects. Especially for interfaces and struc-

tural phenomena on the scale of several nanometers, the calculations of large supercells is inevitable, for which efficient and well-parallelized software packages are required.

## 4. Benchmarks and scaling behaviour

The example simulations shown above were of very different complexity. Let us here briefly discuss the performance of the small GaP supercell (Fig. 1) and the GaP/GaAs heterojunction shown in Fig. 3. We will also show, how STEMsalabim scales with increasing number of computation cores.

As hinted in Section 1, the number of individual multislice simulations grows with number of frozen lattice configurations to account for TDS and required defocus values  $\Delta f$  for inclusion of chromatic aberration effects. The performance of a single multislice simulation depends linearly on the number of slices and on the size  $N \times M$  of the  $k$ - and real-space grid<sup>1</sup>, on which all wave functions are discretized.

The 300 K GaP simulation shown in Fig. 1 had a supercell size of  $9 \times 9 \times 92$  unit cells. As the slice thickness  $dz$  was chosen to match the length of one unit cell (which is a reasonable thickness for [001] beam directions in crystals), the sample was split into 92 slices. A grid density of  $360 \text{ nm}^{-1}$  was used, resulting in a grid size of  $1766 \times 1766$ . A total of  $65 \times 65$  probe positions were simulated (the scanning area was a center square of  $3 \times 3$  unit cells), requiring an overall number of about 443 k multislice simulations for the room temperature simulation (15 frozen lattice configurations for each of the 7 defocus values). The simulation was carried out in parallel on  $7 \times 96$  computation cores with a clock speed of 2.3 GHz each and finished within a runtime of approximately 20 h. All wave functions of the simulation were single precision complex-valued matrices of size  $1766 \times 1766$ . As the transmission functions  $t_s(x, y)$  of all slices are cached (see Section 2.2), the required memory per machine involved in the computation was about 2.2 GB. The majority of simulation time is spent calculating (inverse) fourier transforms, of which for the GaP example more than 50 million are required.

The GaP simulation discussed above is small enough to still run in reasonable time on a desktop computer. In contrast, this becomes almost impossible for the heterojunction shown in Fig. 3. Here, a grid of size  $20118 \times 492$  was used, resulting in much more expensive fourier transforms. The number of scan points calculated was  $2012 \times 29$  and the sample was split into 46 slices. In total, including TDS and defocus series, over 4 million multislice simulations were executed, with required memory of about 5 GB per MPI process. The simulation finished within 4 days and 10 h running in parallel on  $7 \times 256$  2.3 GHz CPUs.

As mentioned before, frozen lattice multislice simulations are parallelizable very efficiently, because the individual multislice simulations are independent from each other. The scaling behaviour with increasing number of processors should therefore be close to linear until the managing master process (I/O or post-processing) becomes the bottle neck of the simulation. We checked the scaling behaviour of STEMsalabim by plotting runtime against number of processing units for single and double precision calculations. The results are displayed in Fig. 4. It is clearly visible that the speedup with increasing number of worker processors is linear, which is the desired behaviour. Double precision calculations are considerably slower than single precision, which is due to slower double precision FFTs.

<sup>1</sup> The precise scaling of the FFTs is governed by the FFT library used and usually depends on how well  $N$  and  $M$  can be factorized.

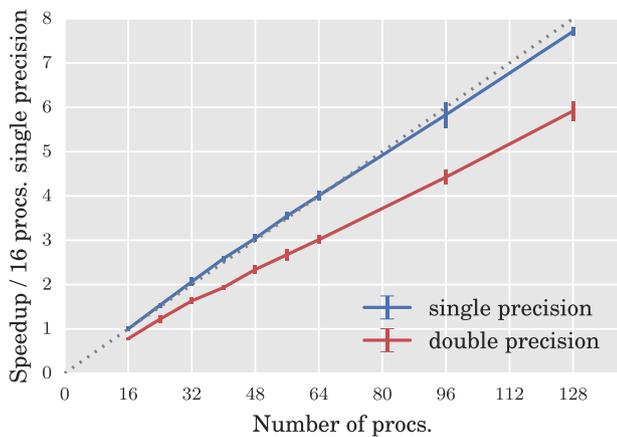


Fig. 4. Speedup relative to a single precision calculation on 16 processors for single and double precision calculations up to 128 processors.

## 5. Conclusion

We have presented a new implementation of the multislice algorithm for STEM image simulations, including chromatic aberrations via defocus series and TDS via the frozen lattice approximation, that is written specifically to run on highly parallelized HPC clusters. We briefly repeated the multislice algorithm and its implementation in Sections 2 and 2.2 and showed example calculations in Section 3. Performance, memory requirements, and scaling with number of processors was briefly discussed in Section 4. We have shown that our code runs successfully in parallel on computing clusters and have shown its computer time and memory efficiency. We have also shown, that the code scales well with increasing number of processors taking part in the simulations.

## Acknowledgements

We would like to acknowledge the creators of STEMsim [13], which we used as a reference implementation to test STEMsalabim. Once again, we would like to highlight the book *Advanced Computing in Electron Microscopy* of Kirkland [1] for its detailed description of the implementation of multislice algorithms.

Support from the German Science foundation (DFG) in the framework of the Research Training Group *Functionalization of Semiconductors* (GRK 1782) and the CRC 1083 *Structure and Dynamics of Internal Interfaces* is gratefully acknowledged.

The STEMsalabim code is available for free on <http://www.online.uni-marburg.de/stemsalabim/>.

## References

- [1] E.J. Kirkland, *Advanced Computing in Electron Microscopy*, 2010, doi:10.1007/978-1-4419-6533-2.
- [2] S.J. Pennycook, D.E. Jesson, High-resolution Z-contrast imaging of crystals, *Ultramicroscopy* 37 (1–4) (1991) 14–38, doi:10.1016/0304-3991(91)90004-P.
- [3] A. Rosenauer, K. Gries, K. Müller, A. Pretorius, M. Schowalter, A. Avramescu, K. Engl, S. Lutgen, Measurement of specimen thickness and composition in  $\text{Al}_x\text{Ga}_{1-x}\text{N}/\text{GaN}$  using high-angle annular dark field images, *Ultramicroscopy* 109 (9) (2009) 1171–1182, doi:10.1016/j.ultramic.2009.05.003.
- [4] V. Grillo, E. Carlino, F. Glas, Influence of the static atomic displacement on atomic resolution Z-contrast imaging, *Phys. Rev. B* 77 (5) (2008) 1–6, doi:10.1103/PhysRevB.77.054103.

- [5] V. Grillo, The effect of surface strain relaxation on HAADF imaging, *Ultramicroscopy* 109 (12) (2009) 1453–1464, doi:10.1016/j.ultramic.2009.07.010.
- [6] P. Goodman, a.F. Moodie, Beam wave functions in electron scattering by the multi-Slice method, *Acta Crystallogr. Sect. A* 30 (2) (1974) 280–290, doi:10.1107/S056773947400057X.
- [7] J.M. Cowley, a.F. Moodie, The scattering of electrons by atoms and crystals. I. A new theoretical approach, *Acta Crystallogr.* 10 (1957) 609–619, doi:10.1107/S0365110X57002194.
- [8] R.F. Loane, P. Xu, J. Silcox, Thermal vibrations in convergent-beam electron diffraction, *Acta Crystallogr. Sect. A* 47 (3) (1991) 267–278, doi:10.1107/S0108767391000375.
- [9] Rolf Erni, High-resolution transmission electron microscopy, in: *Aberration-Corrected Imaging Transmission Electron Microscopy*, Imperial College Press, 2010, pp. 13–43, doi:10.1142/9781848165373\_0002.
- [10] C. Dwyer, C. Maunders, C.L. Zheng, M. Weyland, P.C. Tiemeijer, J. Etheridge, Sub-0.1 nm-resolution quantitative scanning transmission electron microscopy without adjustable parameters, *Appl. Phys. Lett.* 100 (19) (2012) 0–4, doi:10.1063/1.4711766.
- [11] K. Kuramochi, T. Yamazaki, Y. Kotaka, M. Ohtsuka, I. Hashimoto, K. Watanabe, Effect of chromatic aberration on atomic-resolved spherical aberration corrected STEM images, *Ultramicroscopy* 110 (1) (2009) 36–42, doi:10.1016/j.ultramic.2009.09.003.
- [12] A. Beyer, R. Straubinger, J. Belz, K. Volz, Local sample thickness determination via scanning transmission electron microscopy defocus series, *J. Microsc.* 262 (2) (2016) 171–177, doi:10.1111/jmi.12284.
- [13] A. Rosenauer, M. Schowalter, *STEMSIM—a New Software Tool for Simulation of STEM HAADF Z-Contrast Imaging*, Springer Netherlands, Dordrecht, pp. 170–172, doi:10.1007/978-1-4020-8615-1\_36.
- [14] I. Lobato, D. Van Dyck, MULTTEM: A new multislice program to perform accurate and fast electron diffraction and imaging simulations using graphics processing units with CUDA, *Ultramicroscopy* 156 (2015) 9–17, doi:10.1016/j.ultramic.2015.04.016.
- [15] Total Resolution, MacTempasX. <http://www.totalresolution.com/MacTempasX.htm>
- [16] J.M. Zuo, J.C. Mabon, Web-based electron microscopy application software: Web-EMAPS, *Microsc. Microanal.* 10 (S02) (2004) 1000–1001, doi:10.1017/S1431927604884319.
- [17] H.R. Inc., xHREM. <https://www.hremresearch.com/Eng/simulation.html>
- [18] V. Grillo, E. Rotunno, STEM\_CELL: a software tool for electron microscopy: Part I-simulations, *Ultramicroscopy* 125 (2013) 97–111, doi:10.1016/j.ultramic.2012.10.016.
- [19] J. Barthel, Dr. Probe - STEM multislice image calculation program. <http://www.er-c.org/barthel/drprobe/>
- [20] Christoph Koch, *Determination of Core Structure Periodicity and Point Defect Density along Dislocations*, Arizona State University, 2002 Dissertation.
- [21] C. Dwyer, Simulation of scanning transmission electron microscope images on desktop computers, *Ultramicroscopy* 110 (3) (2010) 195–198, doi:10.1016/j.ultramic.2009.11.009.
- [22] L.J. Allen, A.J. D'Alfonso, S.D. Findlay, Modelling the inelastic scattering of fast electrons, *Ultramicroscopy* 151 (2015) 11–22, doi:10.1016/j.ultramic.2014.10.011.
- [23] J. Pizarro, P.L. Galindo, E. Guerrero, A. Yáñez, M.P. Guerrero, A. Rosenauer, D.L. Sales, S.I. Molina, Simulation of high angle annular dark field scanning transmission electron microscopy images of large nanostructures, *Appl. Phys. Lett.* 93 (15) (2008), doi:10.1063/1.2998656.
- [24] M. Haider, S. Uhlemann, J. Zach, Upper limits for the residual aberrations of a high-resolution aberration-corrected STEM, *Ultramicroscopy* 81 (3–4) (2000) 163–175, doi:10.1016/S0304-3991(99)00194-1.
- [25] P.A. Doyle, P.S. Turner, Relativistic HartreeFock X-ray and electron scattering factors, *Acta Crystallogr. Sect. A* 24 (3) (1968) 390–397, doi:10.1107/S0567739468000756.
- [26] P.A. Doyle, J.M. Cowley, *Scattering Factors for the Diffraction of Electrons by Crystalline Solids*, International Tables for Crystallography, Kynoch Press, Birmingham, 1974.
- [27] I. Lobato, D. Van Dyck, An accurate parameterization for scattering factors, electron densities and electrostatic potentials for neutral atoms that obey all physical constraints, *Acta Crystallogr. Sect. A* 70 (2014) 636–649, doi:10.1107/S205327331401643X.
- [28] T. Hoeffer, J. Dinan, D. Buntinas, P. Balaji, B. Barrett, R. Brightwell, W. Gropp, V. Kale, R. Thakur, MPI + MPI: a new hybrid approach to parallel programming with MPI plus shared memory, *Computing* 95 (12) (2013) 1121–1136, doi:10.1007/s00607-013-0324-2.
- [29] A. Beyer, Influence of spatial and temporal coherences on atomic resolution high angle annular dark field imaging, *Ultramicroscopy* 169 (2016) 1–10, doi:10.1016/j.ultramic.2016.06.006.
- [30] O. Rubel, I. Németh, W. Stolz, K. Volz, Modeling the compositional dependence of electron diffraction in dilute GaAs- and GaP-based compound semiconductors, *Phys. Rev. B* 78 (7) (2008) 1–8, doi:10.1103/PhysRevB.78.075207.